We claim:

1. A computer system for generating a random output stream of bits, the system comprising:

an initial evolving state produced from one or more initial keys;

one or more round functions, each round function being part of a step in a sequence of steps, each step applying the respective round function to a current evolving state to produce a respective new evolving state for processing by the next step in the sequence, the initial evolving state processed by the first step in the sequence; and

one or more mask tables produced from one or more of the initial keys, each of the mask tables having one or more masks, one or more of the masks being combined, in each respective step, with the respective new evolving state in a combination operation to create a respective step output, the random output stream being a concatenation of all the respective step outputs, and one or more of the masks in the mask tables being replaced by one or more replacement masks after a number of combination operations, the replacement masks not being linear combinations of prior masks.

2. A computer system, as in claim 1, where the number of combination operations before the mask is replaced by the replacement mask is greater than 1.

3. A computer system, as in claim 1, where the number of combination operations before the mask is replaced by the replacement mask is 16.

4. A computer system, as in claim 1, where one or more of the masks is used in more than one of the combination operations before the mask is replaced by the replacement mask.

5. A computer system, as in claim 1, where two or more tables are produced from the initial keys and one or more mask from each table is used in the combination operation.

6. A computer system, as in claim 5, where the masks from the tables are used in the combination operation in an order.

7. A computer system, as in claim 6, where the order is determined by a value of the respective new evolving state.

8. A computer system, as in claim 5, where the masks from the tables are used in the combination operation in a lexicographical order.

9. A computer system for generating a random output stream of bits, the system comprising:

an initial evolving state produced from one or more initial keys;

one or more round functions, each round function being part of a step in a sequence of steps, each step applying the respective round function to a current evolving state to produce a respective new evolving state for processing by the next step in the sequence, the initial evolving state processed by the first step in the sequence; and

two or more mask tables produced from one or more of the initial keys, each of the mask tables having one or more masks, one or more of the masks from each table being combined, in each respective step, with the respective new evolving state in a combination operation to create a respective step output, the random output stream being a concatenation of all the respective step outputs.

10. A computer system, as in claim 9, where the masks from the tables are used in the combination operation in an order.

11. A computer system, as in claim 9, where the masks from the tables are used in the combination operation in a lexicographical order.

12. A computer system, as in claim 9, where the order is determined by a value of the respective new evolving state.

13. A computer system, as in claim 1, where the round function is a non linear permutation.

14. A computer system, as in claim 13, where the non linear permutation includes any one or more of the following: a substitution-perumutation network and a Feistel ladder.

15. A computer system, as in claim 13, where the non linear permutation performed by the round function comprises the following steps:

dividing the current evolving state into a first part and one or more second parts;

applying a first non linear function to the first part to create a first part first result;

applying one or more second non linear functions to the first part to create one or more first part second results;

combining one or more first part second results to one or more of the second parts to create one or more respective interim second parts; and

concatenating the first part first result and the interim second parts to create a new evolving state.

16. A computer system, as in claim 13, where the non linear permutation performed by the round function comprises the following steps:

dividing the current evolving state into a first part and a second part;

applying a first non linear function to the first part to create a first part first result:

applying a second non linear function to the first part to create a first part second result;

combining the first part second result to the second part to create a respective interim second part;

applying the first non linear function to the interim second part to create a final first result;

applying the second non linear function to the interim second part to create an interim second part second result;

combining the interim second part second result with the first part second result to create a final second result; and

concatenating the final first result and the final second result to create a new evolving state.

17. A method for generating a random output stream of bits comprising the steps of:

A. producing a current evolving state from one or more initial keys;

B. producing one or more mask tables from one or more of the initial keys, each of the mask tables having one or more masks;

C. applying a round function to a current evolving state to produce a respective new evolving state;

D. replacing the current evolving state with the new evolving state;

E. combining one or more of the masks with the current evolving state in a combination operation to create a respective step output;

F. replacing one or more of the masks in the mask tables by one or more replacement masks after a number of combination operations, the replacement masks not being linear combinations of prior masks.

G. repeating steps C through F one or more times; and

H. concatenating all the respective step outputs to create the random output stream.

18. A method, as in claim 17, where the round function is non linear perturbation method comprising the steps of:

dividing the current evolving state into a first part and one or more second parts;

applying a first non linear function to the first part to create a first part first result;

applying one or more second non linear functions to the first part to create one or more first part second results;

combining one or more first part second results to one or more of the second parts to create one or more respective interim second parts; and

concatenating the first part first result and the interim second parts to create a new evolving state.

19. A computer program product having a stored method for generating a random output stream of bits, the method comprising the steps of:

A. producing a current evolving state from one or more initial keys;

B. producing one or more mask tables from one or more of the initial keys, each of the mask tables having one or more masks;

C. applying a round function to a current evolving state to produce a respective new evolving state;

D. replacing the current evolving state with the new evolving state;

E. combining one or more of the masks with the current evolving state in a combination operation to create a respective step output;

F. replacing one or more of the masks in the mask tables by one or more replacement masks after a number of combination operations, the replacement masks not being linear combinations of prior masks.

G. repeating steps C through F one or more times; and

H. concatenating all the respective step outputs to create the random output stream.

20. A computer system for generating a random output stream of bits, the system comprising:

A. means for producing a current evolving state from one or more initial keys;

B. means for producing one or more mask tables from one or more of the initial keys, each of the mask tables having one or more masks;

C. means for applying a round function to a current evolving state to produce a respective new evolving state;

D. means for replacing the current evolving state with the new evolving state;

E. means for combining one or more of the masks with the current evolving state in a combination operation to create a respective step output;

F. means for replacing one or more of the masks in the mask tables by one or more replacement masks after a number of combination operations, the replacement masks not being linear combinations of prior masks.

G. means for repeating steps C through F one or more times; and

H. means for concatenating all the respective step outputs to create the random output stream.